GPBR Exercise 3

Name: Soi Zhi Wen

Matriculation Number: 22-132-245

1. $A_{g_1} = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix}$

   $A_{g_2} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix}$

   The eigendecomposition of g_1:
   $[\ 2.48119430e+00\ \ -2.00000000e+00\ \ -1.17008649e+00\ \ 8.84119150e-17\ \ 6.88892183e-01]$
   $[[-5.29899099e-01\ \ -5.00000000e-01\ \ -4.32486630e-01\ \ 5.00000000e-01\ \ -1.79338395e-01]$
   $[-4.27132287e-01\ \ -4.37318244e-17\ \ 7.39238740e-01\ \ 1.93496901e-16\ \ -5.20657368e-01]$
   $[-5.29899099e-01\ \ 5.00000000e-01\ \ -4.32486630e-01\ \ -5.00000000e-01\ \ -1.79338395e-01]$
   $[-3.57751240e-01\ \ -5.00000000e-01\ \ 1.99294651e-01\ \ -5.00000000e-01\ \ 5.76450945e-01]$
   $[-3.57751240e-01\ \ 5.00000000e-01\ \ 1.99294651e-01\ \ 5.00000000e-01\ \ 5.76450945e-01]]$

   The eigendecomposition of g_2:
   $[-1.68133064\ \ 0.35792637\ \ 3.32340428\ \ -1.\ \ \ \ -1.\ \ \ ]$
   $[[\ 5.59032552e-01\ \ -7.70242078e-01\ \ 3.06936062e-01\ \ -9.59450110e-17\ \ 6.17375934e-17]$
   $[-4.69959281e-01\ \ -1.37844975e-01\ \ 5.10036310e-01\ \ 6.11078733e-01\ \ -3.93209922e-01]$
   $[-4.69959281e-01\ \ -1.37844975e-01\ \ 5.10036310e-01\ \ -6.11078733e-01\ \ 3.93209922e-01]$
   $[\ 3.50541834e-01\ \ 4.29374351e-01\ \ 4.39042241e-01\ \ -3.55784741e-01\ \ -5.87695463e-01]$
   $[\ 3.50541834e-01\ \ 4.29374351e-01\ \ 4.39042241e-01\ \ 3.55784741e-01\ \ 5.87695463e-01]]$

   The resulting $\overline{U}_{g_1}\overline{U}_{g_1}^T$ matrix:
   $[[-0.09727862\ \ 0.23285414\ \ -0.96490171\ \ 0.06279861\ \ -0.04040893]$
   $[-0.51449592\ \ 0.31619855\ \ 0.10155012\ \ 0.05024537\ \ 0.78430042]$
   $[-0.24621288\ \ 0.46197919\ \ 0.19870651\ \ 0.71601645\ \ -0.46073404]$
   $[\ 0.51449592\ \ -0.31619855\ \ -0.10155012\ \ 0.66132411\ \ 0.3910905\ ]$
   $[\ 0.63285385\ \ 0.72965138\ \ 0.09410395\ \ -0.20857277\ \ 0.13421001]]$

   Assignment found by LSAP solver:
   $[0\ 1\ 2\ 3\ 4] = [1\ 2\ 3\ 4\ 5]$
   $[1\ 4\ 2\ 3\ 0] = [b\ e\ c\ d\ a]$
   This indicates that
   node 1 in g_1 is matched with node b in g_2,
   node 2 in g_1 is matched with node e in g_2,
   node 3 in g_1 is matched with node c in g_2,
   node 4 in g_1 is matched with node d in g_2,

node 5 in g_1 is matched with node a in g_2.

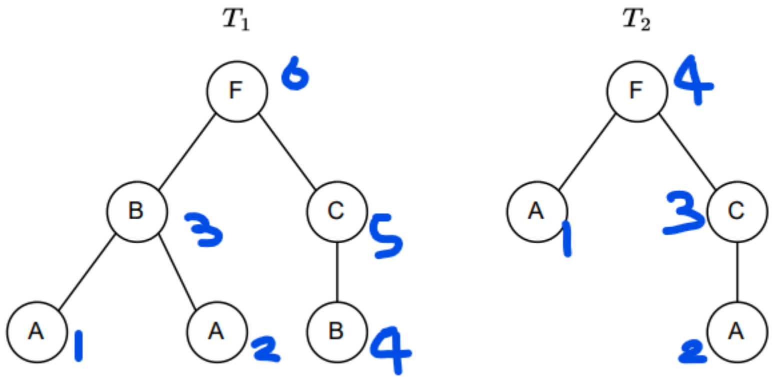
2.

Spectral Graph Matching	**Continuous Graph Matching**
1. Compute the Laplacian matrix for each graph.	1. Compute the distance metric between each pair of nodes in each graph.
2. Compute the first k eigenvectors of each Laplacian matrix.	2. Solve the continuous optimization problem to find the best correspondence between the graphs.
3. Construct a matrix Y for each graph, where the columns are the k eigenvectors.	3. Use the optimal correspondence to transform one graph into the other.
4. Compute the orthogonal Procrustes transformation between the two Y matrices.	
5. Apply the transformation to one of the Y matrices.	
6. Compute the matrix product of the transformed Y matrix and the other original Y matrix.	

<table>
<tr><th>Spectral Graph Matching</th><th>Continuous Graph Matching</th></tr>
<tr><td colspan="2" align="center">Similarities</td></tr>
<tr><td colspan="2">Both approaches aim to find a permutation matrix that maps nodes in one graph to nodes in another graph, such that the two graphs are as similar as possible.</td></tr>
<tr><td colspan="2">Both approaches use optimization techniques to find the best permutation matrix that minimizes a given objective function.</td></tr>
<tr><td colspan="2">Both approaches are based on the idea of embedding graphs into a lower-dimensional space, where matching can be performed more efficiently.</td></tr>
<tr><td colspan="2" align="center">Differences</td></tr>
<tr><td>Uses eigenvectors of the graph Laplacian to embed the graphs into a lower-dimensional space.</td><td>Uses a mapping function to embed the graphs into a continuous space.</td></tr>
<tr><td>Requires that the two graphs being matched have the same number of nodes.</td><td>Can handle graphs of different sizes.</td></tr>
</table>

More sensitive to noise in the eigenvectors.	Handle noisy data more effectively since it allows for a smooth mapping between the two graphs.

3.



	∅	$T_2[1]$	$T_2[2]$	$T_2[3]$	$T_2[4]$
∅	0	1	2	3	4
$T_1[1]$	1	0	1	2	3
$T_1[2]$	2	1	0	1	2
$T_1[3]$	3	2	1	1	2
$T_1[4]$	4	3	2	2	2
$T_1[5]$	5	4	3	2	3
$T_1[6]$	6	5	4	3	2

4.